A Note on Neighbor Finding in Images Represented by Quadtrees [1]

Ye-In Chang, and Hue-Ling Chen

Dept. of Computer Science and Engineering
National Sun Yat-Sen University
Kaohsiung, Taiwan
Republic of China
{E-mail: changyi@cse.nsysu.edu.tw}
{Tel: 886-7-5252000 (ext. 4334)}
{Fax: 886-7-5254301}

Abstract

Neighbor finding is one of the most important spatial operations in the field of spatial data structures. A block B is said to be the neighbor of another block A, if block B has the same property as block A has and covers an equal sized neighbor of block A. Jozef Voros has proposed a neighbor finding strategy on images represented by quadtrees, in which the four equal sized neighbors (the east, west, north, and south directions) of block A can be found. However, based on Voros's strategy, the case that the nearest neighbor occurs in the diagonal directions (the northeast, northwest, southeast, and southwest directions) will be ignored. Therefore, in this paper, we first show the missing case. Next, we show how to find the missing cases.

(*Key Words:* inner neighbors, locational codes, neighbor finding, outer neighbors, quadtrees.)

[1] This research was supported in part by the National Science Council of Republic of China under Grant No. NSC-90-2213-E-110-027.

1 Introduction

Recently, spatial database systems [4] are popular and important for public administration, science, and business. There are many applications that deal with spatial data, including Geographic Information Systems (GIS), and multimedia databases. Spatial data consists of spatial objects made up of points, lines, regions, rectangles, surfaces, volumes, and even data of higher dimension [2, 5, 12]. High-dimensional datasets are typically mapped into collections of points in the coordinate space, and the nearest neighbor queries are the most kinds of queries. Such a query like retrieving objects whose representative points are closest to the point representing the query point is very common. For example, the typical queries are "Find the 10 cities that are the nearest to Madison" in a GIS system and "Ask for images similar to the query image" in the multimedia database. When looking for the nearest neighbors in the spatial database, it can be modeled as finding points that are close to each other in the image. For those applications mentioned above, the data volume is extremely high, the spatial objects show a very complex structure and the computation of spatial operators is time-intensive. The computation of spatial operators is more difficult than that of the non-spatial counterparts because of no total ordering by spatial proximity among spatial objects.

Neighbor finding [1, 6, 7, 8, 9, 10] is one of the most important spatial operations about spatial proximity in the field of spatial data structures. Block B is said to be a neighbor of block A in one direction if block B is adjacent to block A. Jozef Voros [13] has proposed a new view on the neighbor finding problem in *linear quadtrees* [3, 8, 9, 11, 13], which focused on distinguishing between the *inner* and *outer* neighbors and searching neighbors in two main (horizontal and vertical) directions. Block B is an inner neighbor of block A if blocks A and B belong to the same quadrant. Otherwise, block B is an outer neighbor of block A.

Take Figure 1 as an example. Given the query point q in block A, the inner neighbors of block A are block B in the horizontal direction and block C in the vertical direction. There exist two outer neighbors of block A: one is block I in the horizontal direction and the other is block G in the vertical direction. The quadtree corresponding to Figure 1, based on the order defined in Figure 2, is shown in Figure 3.

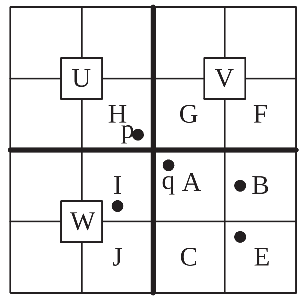

Figure 1: Example 1

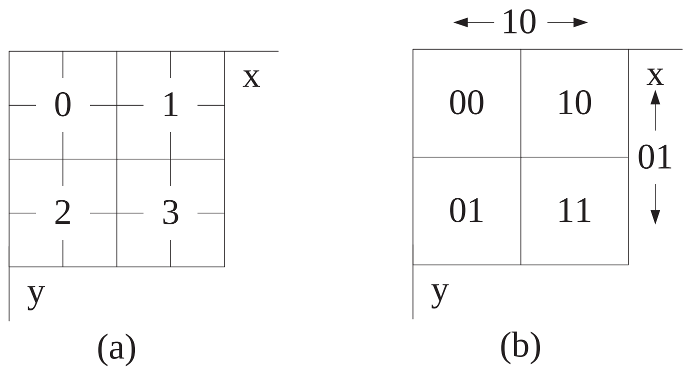

Figure 2: Quadrant's order and directions: (a) the decimal values; (b) the binary codes.

Given the binary locational code of the block in which the query point is located, according to Figure 2, the proposed approach uses only simple algebraic operations to generate equal sized neighbors for the query block. Although blocks B, C, G, and I in Figure 1 can be generated by Voros's strategy [13], the nearest neighbor to the query point q in block A is point p in block H. In this case, block H, which is the neighbor in the diagonal direction to block A is ignored. Similarly, blocks E, F, and J which are in the diagonal directions can not be obtained based on Voros's strategy. Therefore, in this paper, we first briefly describe some formulas used in the neighbor generating method proposed by Jozef Voros [13]. Then, we give a missing case. Finally, we show how to find the missing cases.

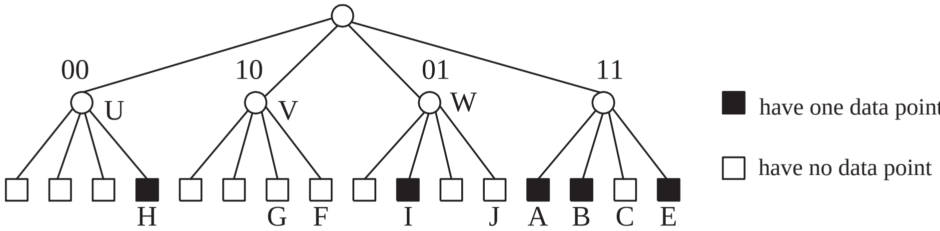

Figure 3: The corresponding quadtree for example 1

2 Neighbor Generation

Basically, there are two kinds of equal sized neighbors: inner and outer neighbors. Take Figure 1 for example, it is a $(2^2 \times 2^2)$- dimensional binary image. According to Figure 2-(a), the locational code of block A is represented by quaternary digits as $(\underline{3}\ \underline{0})$. In order to simplify the process of neighbor finding, the locational code of block A is represented in the binary form as $(\underline{x_0, y_0}, \underline{x_1, y_1})$=$(\underline{1, 1},\ \underline{0, 0})$ according to Figure 2-(b).

2.1 Equal Sized Neighbors

For the equal sized inner neighbors of block A in Figure 1, there exists one, block B in the direction $D = [1, 0]$ (left and right) according to Figure 2-(b). Its locational code is $(\underline{x_0, y_0}, \underline{x_1, y_1}) = (\underline{1, 1}, \underline{1^*, 0}) = (\underline{3}\ \underline{1})$, where * means the changed digit, and it can be obtained by using equations (7), (8), and (9). In each of these equations, the right part of the equation means the element (x_i or y_i) of block A, and the left part means the element ($\underline{x}_i$ or $\underline{y}_i$) of its neighbor. Similarly, the inner neighbor in the direction $D = [0, 1]$ (up and down) is block C whose locational code is $(\underline{x_0, y_0}, \underline{x_1, y_1})$=$(\underline{1, 1},\ \underline{0, 1^*}) = (\underline{3}\ \underline{2})$, and it can be obtained by using equations (7), (8), and (10).

Equations (7) – (10):

$$\underline{x}_j = x_j, \qquad j = 0, 1, \ldots, n-1 \tag{7}$$

$$\underline{y}_j = y_j, \qquad j = 0, 1, \ldots, n-1 \tag{8}$$

$$\underline{x}_n = comp(x_n), \qquad \underline{y}_n = y_n, \qquad for D = [1, 0] \tag{9}$$

$$\underline{x}_n = x_n, \qquad \underline{y}_n = comp(y_n), \qquad for D = [0, 1] \tag{10}$$

where *comp()* means the complement and n means the *level* number, which means the level in the corresponding quadtree. The block A and its inner neighbors always differ in one bit [13].

For the equal sized outer neighbors, it can be classified into two cases: one is along x axis (in $D = [1,0]$) and the other is along y axis (in $D = [0,1]$). Take Figure 1 as an example, the equal sized outer neighbor of block A in $D = [1,0]$ can be obtained as $(\underline{x_0, y_0}, \underline{x_1, y_1}) = (\underline{0^*, 1}, \underline{1^*, 0}) = (\underline{2}\ \underline{1})$ by using equations (11) – (14). That is block I shown in Figure 1.

Equations (11) – (14):

Let $k > 0$ be such an integer that

$$x_n = x_{n-1} = \ldots = x_{n-k+1} \neq x_{n-k}. \tag{11}$$

Then, if $k \leq n$, there exists one *equal sized outer neighbor* in the direction $D = [1,0]$ and its binary locational code $\{\underline{x}_i, \underline{y}_i\}$ is computed as follows [13]:

$$\underline{x}_{n-i} = comp(x_{n-i}), \qquad i = 0, 1, \ldots, k \tag{12}$$

$$\underline{x}_i = x_i, \qquad i = 0, 1, \ldots, n-k-1 \tag{13}$$

$$\underline{y}_i = y_i, \qquad i = 0, 1, \ldots, n \tag{14}$$

For the equal sized outer neighbor of block A in $D = [0,1]$, it is block G whose the locational code is $(\underline{x_0, y_0}, \underline{x_1, y_1}) = (\underline{1, 0^*}, \underline{0, 1^*}) = (\underline{1}\ \underline{2})$ by using equations (15) – (18).

Equations (15) – (18):

If $k > 0$ is an integer such that

$$y_n = y_{n-1} = \ldots = y_{n-k+1} \neq y_{n-k}. \tag{15}$$

Then, if $k \leq n$, there exists one *equal sized outer neighbor* in the direction $D = [0,1]$ and its binary locational code $\{\underline{x}_i, \underline{y}_i\}$ is computed as follows [13]:

$$\underline{y}_{n-i} = comp(y_{n-i}), \qquad i = 0, 1, \ldots, k \tag{16}$$

$$\underline{y}_i = y_i, \qquad i = 0, 1, \ldots, n-k-1 \tag{17}$$

$$\underline{x}_i = x_i, \qquad i = 0, 1, \ldots, n \tag{18}$$

To obtain the locational code of its outer neighbors, more digits of the locational code of block A have to be changed. The number of required changes is determined by the locational code of block A.

2.2 Larger Sized Neighbors

The larger sized neighbor of block A in Figure 1 can be derived from the locational code of the equal sized outer neighbor, i.e., block V derived from block G and block W derived from block I, where block G and I are equal sized outer neighbors of block A in two directions. The binary locational codes $((\underline{x}_0, \underline{y}_0, \underline{x}_1, \underline{y}_1, \ldots, \underline{x}_{n-j}, \underline{y}_{n-j}), 1 \leq j \leq k)$ of the larger sized neighbors are generates from the code $(\{x_i, y_i\}, 0 \leq i \leq n)$ of the corresponding equal sized outer neighbor by successive cancelling the least significant digits up to the $(n-k)$th level, where k is given by equation (11) or equation (15). That is, the locational code of block V is $(\underline{x_0, y_0}) = (\underline{1, 0}) = (\underline{1})$, and the locational code of block W is $(\underline{x_0, y_0}) = (\underline{0, 1}) = (\underline{2})$.

2.3 Smaller Sized Neighbors

The smaller sized neighbors of node A in Figure 1 are the pairs of contiguous quadrants, subquadrants, ..., of its equal sized neighbor sharing a common side with A. The inner and outer smaller sized neighbors can be distinguished in two main directions. They are derived from the corresponding equal sized inner or outer neighbor whose locational code is $\{x_i, y_i\}, 0 \leq i \leq n$. The smaller sized neighbors at the $(n+k)$th level in the direction $D = [0, 1]$ are all the nodes ${}^iN, i = 1, 2, \ldots, 2^k$, with the following binary locational codes, i.e.,

$$ {}^iN = (\underline{x}_0, \underline{y}_0, \underline{x}_1, \underline{y}_1, \ldots \underline{x}_n, \underline{y}_n, \underline{x}_{n+1}, \underline{y}_{n+1}, \underline{x}_{n+2}, \underline{y}_{n+2}, \ldots \underline{x}_{n+k}, \underline{y}_{n+k}) \tag{23} $$

$$ \underline{x}_{n+j} = d, \qquad j = 1, 2, \ldots, \tag{24} $$

where d is 0 or 1, and for all outer neighbors,

$$ \underline{y}_{n+j} = \underline{y}_n; \tag{25} $$

while for all inner neighbors,

$$ \underline{y}_{n+j} = comp(\underline{y}_n). \tag{26} $$

Analogously, the locational codes of smaller sized neighbors in the direction $D = [1, 0]$ can be derived by interchanging the role of $x's$ and $y's$ in equations (24)–(26). Take Figure 4 as

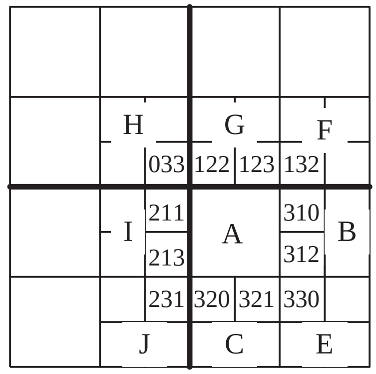

Figure 4: Another representation of Example 1

an example, it is another representation of Example 1 in Figure 1. The one pair of smaller sized inner neighbor of block A ($\underline{3}\ \underline{0}$) in $D = [0, 1]$ can be obtained as $(\underline{x_0, y_0}, \underline{x_1, y_1}, \underline{x_2, y_2}) = (\underline{1, 1}, \underline{0, 1}, \underline{0^*, 0^*}) = (\underline{3}\ \underline{2}\ \underline{0})$ and $(\underline{x_0, y_0}, \underline{x_1, y_1}, \underline{x_2, y_2}) = (\underline{1, 1}, \underline{0, 1}, \underline{1^*, 0^*}) = (\underline{3}\ \underline{2}\ \underline{1})$ which are derived from the equal sized inner neighbor C by using equations (23), (24), and (26). The other one pair of smaller sized outer neighbor of block A in $D = [0, 1]$ can be obtained as $(\underline{x_0, y_0}, \underline{x_1, y_1}, \underline{x_2, y_2}) = (\underline{1, 0}, \underline{0, 1}, \underline{0^*, 1^*}) = (\underline{1}\ \underline{2}\ \underline{2})$ and $(\underline{x_0, y_0}, \underline{x_1, y_1}, \underline{x_2, y_2}) = (\underline{1, 0}, \underline{0, 1}, \underline{1^*, 1^*}) = (\underline{1}\ \underline{2}\ \underline{3})$ which are derived from the equal sized outer neighbor G by using equations (23), (24), and (25). In the same way, blocks 310, 312, 211, and 213 which are the smaller sized inner and outer neighbors of block A can be found.

3 A Missing Case

Take Figure 5 as an example: the query point in block 120, based on Voros's strategy [13], the binary form of the block whose locational code is represented by quatered digits, $\underline{1}$ $\underline{2}$ $\underline{0}$, is $(\underline{x_0, y_0}, \underline{x_1, y_1}, \underline{x_2, y_2}) = (\underline{1, 0}, \underline{0, 1}, \underline{0, 0})$. For the first case: equal sized inner neighbors, it has one in the direction $D = [1, 0]$, that is ($\underline{1}\ \underline{2}\ \underline{1}$)= $(\underline{1, 0}, \underline{0, 1}, \underline{1^*, 0})$ obtained by changing the last digit, x_2, of the block 120 from 0 to 1 according to equation (9). In $D = [0, 1]$, it has the other one inner neighbor, that is ($\underline{1}\ \underline{2}\ \underline{2}$)= $(\underline{1, 0}, \underline{0, 1}, \underline{0, 1^*})$ obtained by changing the last digit, y_2, of block 120 from 0 to 1 according to equation (10).

For the second case: equal sized outer neighbors, block 120 has one in the direction $D = [1, 0]$. According to equations (11) – (14), the relation between the digits x_2, x_1, and

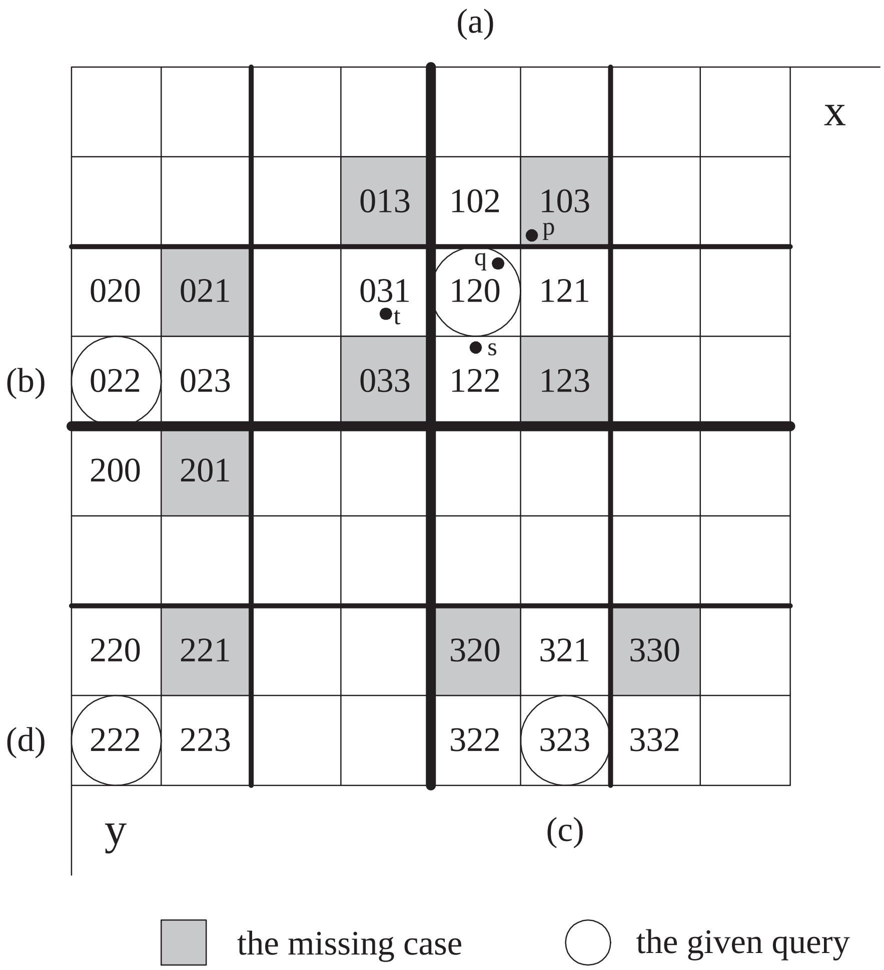

Figure 5: Example 2: (a) case1; (b) case2; (c) case 3; (d) case 4.

x_0 of the locational code of block 120 is: $x_2 = x_1 \neq x_0$. Then, the locational code of the outer neighbor is obtained from that of block 120 after changing three digits: x_2, x_1, and x_0, that is $(\underline{0^*, 0}, \underline{1^*, 1}, \underline{1^*, 0}) = (\underline{0}\ \underline{3}\ \underline{1})$. Similarly, in the direction $D = [0, 1]$, the relation between the digits y_2, y_1, and y_0 of the locational code of block 120 is: $y_2 \neq y_1$. It means that the locational code of the outer neighbor is obtained from that of block 120 after changing two digits: y_2, y_1, according to equations (15) – (18). That is, $(\underline{1, 0}, \underline{0, 0^*}, \underline{0, 1^*}) = (\underline{1}\ \underline{0}\ \underline{2})$. These neighbors are shown in the Figure 5-(a) as the quaternary digits of their locational code. Based on Voros's strategy [13], the nearest neighbor finding process is finished at this point.

However, it shows that some neighbors miss because one of the neighbors which is in the diagonal directions may be the nearest neighbor. In Figure 5, based on Voros's strategy [13], we obtain that the nearest point to the query point q in block 120 is point s in block 122. But the real nearest point to the query point q is point p in block 103 which is in the shaded area and is ignored. So, in fact, we must also search for the nearest neighbor in the four or less than four diagonal directions, like the other degenerated cases when the query point is located in blocks 022, 323, or 222. Besides, the larger and the smaller sized neighbors in the diagonal directions must be considered, when the data points are stored in these different sized blocks, instead of the equal sized blocks. To overcome this problem, we provide an answer. We can find the ignored case by adding some equations to obtain the locational codes of the neighbors in the diagonal directions.

4 The Improved Version of Neighbor Finding

From the above example, we show that the missing case can occur in the neighbor finding based on Voros's strategy [13]. To find this case, we add equations (27) – (37) below to help looking for the equal sized neighbors in the diagonal directions. Given a query point with its located block at the nth level and the binary locational code denoted as $\{x_i, y_i\}$, for the equal sized inner neighbors in the diagonal direction, the binary locational code $\{\underline{x}_i, \underline{y}_i\}$ is given by equations (7), (8), (27) and (28).

$$\underline{x}_j = x_j, \qquad j = 0, 1, \ldots, n-1 \tag{7}$$

$$\underline{y}_j = y_j, \qquad j = 0, 1, \ldots, n-1 \tag{8}$$

$$\underline{x}_n = comp(x_n) \tag{27}$$

$$\underline{y}_n = comp(y_n). \tag{28}$$

For the equal sized outer neighbors in the diagonal direction, it can be classified into three cases.

1. x-outer neighbor's (along x-axis) y-inner neighbor which is along y-axis.

2. y-outer neighbor's (along y-axis) x-inner neighbor which is along x-axis.

3. x-outer neighbor's (along x-axis) y-outer neighbor which is along y-axis or y-outer neighbor's (along y-axis) x-outer neighbor which is along x-axis.

For example, in Figure 5-(a), for the query point q in block 120, block 123 is point q's inner neighbor in the diagonal direction. Block 033 is point q's x-outer neighbor's (block 031 along x-axis) y-inner neighbor (along y-axis). Block 103 is point q's y-outer neighbor's (block 102 along y-axis) x-inner neighbor (along x-axis). Block 013 is point q's x-outer neighbor's (block 031 along x-axis) y-outer neighbor (along y-axis) and it is also point q's y-outer neighbor's (block 102 along y-axis) x-outer neighbor (along x-axis).

First, let's consider the case of x-outer neighbor's (along x-axis, that is in $D = [1, 0]$) y-inner neighbor along y-axis (in $D = [0, 1]$). Let $k > 0$ be such an integer that

$$x_n = x_{n-1} = \ldots = x_{n-k+1} \neq x_{n-k}. \tag{11}$$

Then, if $k \leq n$, there exists one equal sized *x-outer neighbor's y-inner neighbor*, which is the outer neighbor in the diagonal direction and its binary locational code $\{\underline{x}_i, \underline{y}_i\}$ is given by equations (12), (13), (29), and (30).

$$\underline{x}_{n-i} = comp(x_{n-i}), \qquad i = 0, 1, \ldots, k \tag{12}$$

$$\underline{x}_i = x_i, \qquad i = 0, 1, \ldots, n-k-1 \tag{13}$$

$$\underline{y}_i = y_i, \qquad i = 0, 1, \ldots, n-1 \tag{29}$$

$$\underline{y}_n = comp(y_n). \tag{30}$$

Second, let's consider the case of y-outer neighbor's (in $D = [0, 1]$) x-inner neighbor in $D = [1, 0]$. If $k > 0$ is an integer such that

$$y_n = y_{n-1} = \ldots = y_{n-k+1} \neq y_{n-k}. \tag{15}$$

Then, if $k \leq n$, there exists one equal sized *y-outer neighbor's x-inner neighbor*, which is the outer neighbor in the diagonal direction and its binary locational code $\{\underline{x}_i, \underline{y}_i\}$ is given by equations (16), (17), (31), and (32).

$$\underline{y}_{n-i} = comp(y_{n-i}), \qquad i = 0, 1, \ldots, k \tag{16}$$
$$\underline{y}_i = y_i, \qquad i = 0, 1, \ldots, n-k-1 \tag{17}$$
$$\underline{x}_i = x_i, \qquad i = 0, 1, \ldots, n-1 \tag{31}$$
$$\underline{x}_n = comp(x_n). \tag{32}$$

Third, let's consider the *x-outer neighbor's y-outer neighbor* or *y-outer neighbor's x-outer neighbor*. Let $h > 0$ and $k > 0$ be integers such that

$$x_n = x_{n-1} = \ldots = x_{n-h+1} \neq x_{n-h}, \quad and \quad y_n = y_{n-1} = \ldots = y_{n-k+1} \neq y_{n-k}. \tag{33}$$

Then, if $h \leq n$ and $k \leq n$, there exists another one equal sized *outer neighbor's outer neighbor*, which is the outer neighbor in the diagonal direction and its binary loc ational code $\{\underline{x}_i, \underline{y}_i\}$ is given by equations (34), (35), (36), and (37).

$$\underline{x}_{n-i} = comp(x_{n-i}), \qquad i = 0, 1, \ldots, h \tag{34}$$
$$\underline{x}_i = x_i, \qquad i = 0, 1, \ldots, n-h-1 \tag{35}$$
$$\underline{y}_{n-i} = comp(y_{n-i}), \qquad i = 0, 1, \ldots, k \tag{36}$$
$$\underline{y}_i = y_i, \qquad i = 0, 1, \ldots, n-k-1. \tag{37}$$

For example, block ($\underline{1}$ $\underline{2}$ $\underline{0}$) = ($\underline{1, 0}, \underline{0, 1}, \underline{0, 0}$) in Figure 5-(a) has an equal sized inner neighbor in the diagonal direction that is obtained by changing the last two digits 0, 0 of the locational code to 1, 1 according to equations (7), (8), (27), and (28). The locational code of this neighbor is ($\underline{1, 0}, \underline{0, 1}, \underline{1^*, 1^*}$) = ($\underline{1}$ $\underline{2}$ $\underline{3}$). Generally, every block has an inner neighbor in the diagonal direction, and their locational code in the binary form always differ in the last two digits.

Next, block ($\underline{1}$ $\underline{2}$ $\underline{0}$) = ($\underline{1, 0}, \underline{0, 1}, \underline{0, 0}$) in Figure 5-(a) has an equal sized outer neighbor in the diagonal direction which is its x-outer neighbor's (block 031 in $D = [1, 0]$) y-inner neighbor in $D = [0, 1]$. By equation (11), the relation between x_0, x_1, and x_2 is: $x_2 = x_1 \neq x_0$. Then according to equations (12), (13), (29), and (30), block ($\underline{0}$ $\underline{3}$ $\underline{3}$) = ($\underline{0^*, 0}, \underline{1^*, 1}, \underline{1^*, 1^*}$) shown in Figure 5-(a) is obtained as one of the equal sized outer neighbors to block 120 in the diagonal direction. The other equal sized outer neighbor in the diagonal direction is the y-outer neighbor's (block 102 in $D = [0, 1]$) x-inner neighbor in $D = [1, 0]$. By equation (15), the relation between y_0, y_1, and y_2 is: $y_2 \neq y_1$. Then, we can obtain the locational code $(\underline{1, 0}, \underline{0, 0^*}, \underline{1^*, 1^*}) = (\underline{1}\ \underline{0}\ \underline{3})$ according to the equations (16), (17), (31), and (32). Block 103 shown in Figure 5-(a) is one of the equal sized outer neighbors to block 120 in the diagonal direction. According to equation (33), we obtain the equations: $x_2 = x_1 \neq x_0$; $y_2 \neq y_1$; $h = 2$ and $h \leq n$ $(= 2)$; $k = 1$ and $k \leq n$ $(= 2)$. From equations (34) – (37), block $(\underline{0}\ \underline{1}\ \underline{3}) = (\underline{0^*, 0}, \underline{1^*, 0^*}, \underline{1^*, 1^*})$ in Figure 5-(a) is another equal sized outer neighbor to block 120 in the diagonal direction. Finally, we can obtain 8 equal sized neighbors around block 120.

From the above example, we can find the missing case by adding equations from (27) to (37) to find all the equal sized neighbors around the query block. Besides, we need to find the different sized neighbors around the query block because the data points may be stored in these blocks. Based on Voros's strategy [13], the larger sized neighbors of the query block 120 are block 10 which is derived from the code of equal sized outer neighbor 102, block 03 which is derived from the code of equal sized outer neighbor 031, and block 0 which is derived from the code of equal sized outer neighbor 031. Block 01 which is derived from the code of equal sized outer neighbor 013 in the diagonal directions can also be found by using Voros's strategy [13].

The smaller sized inner neighbors of block 120 in Figure 5-(a) are shown as follows by using equations (23), (24), and (26) in Voros's strategy [13].

- In $D = [1, 0]$, blocks 1210 and 1212 in Figure 6-(a) are derived from the equal sized inner neighbor 121 in Figure 5-(a).
- In $D = [0, 1]$, blocks 1220 and 1221 in Figure 6-(a) are derived from the equal sized inner neighbor 122 in Figure 5-(a).

The smaller sized outer neighbors of block 120 in Figure 5-(a) are shown as follows by using equations (23), (24), and (25) in Voros's strategy [13].

- In $D = [1, 0]$, blocks 0311 and 0313 in Figure 6-(a) are derived from the equal sized outer neighbor 031 in Figure 5-(a).

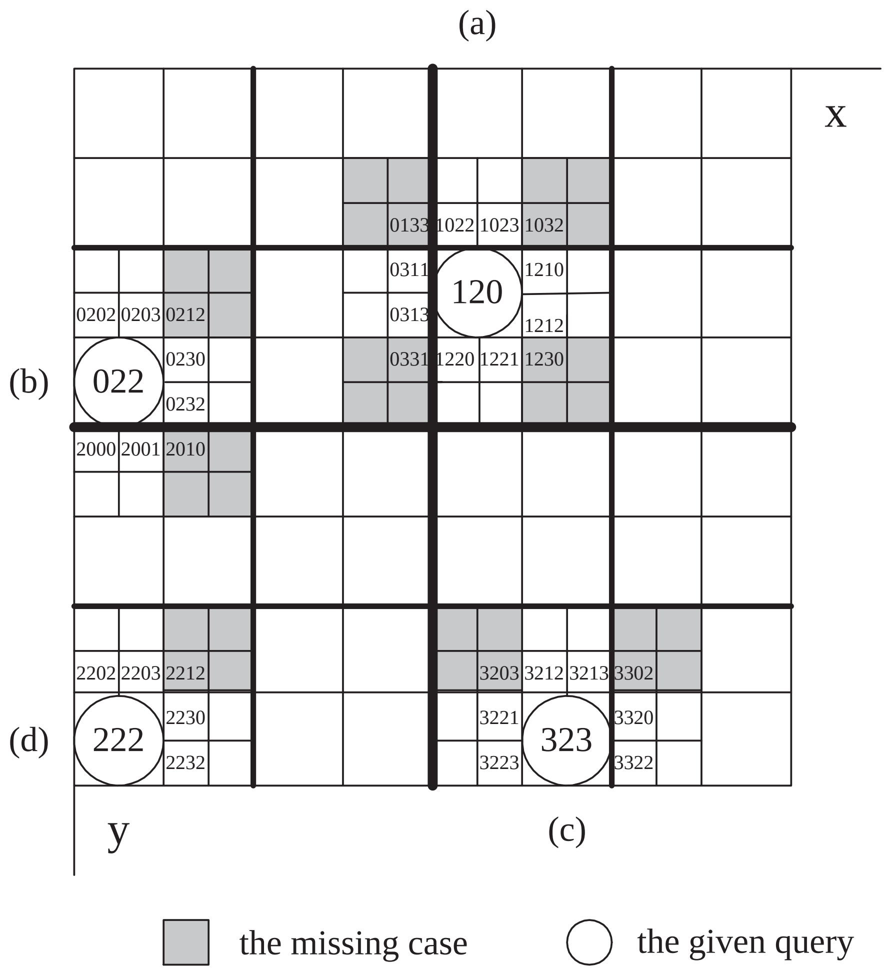

Figure 6: Another representation of Example 2: (a) case1; (b) case2; (c) case 3; (d) case 4.

- In $D = [0, 1]$, blocks 1022 and 1023 in Figure 6-(a) are derived from the equal sized outer neighbor 102 in Figure 5-(a).

But the smaller sized inner or outer neighbors in the diagonal directions are ignored in Voros's strategy [13]. We add some equations to find them which are shown as follows.

The smaller sized inner neighbors at the $(n+k)$th level in the diagonal direction are all the nodes ${}^iN, i = 1, 2, \ldots, 2^k$, with the following binary locational codes derived from the code $(x_0, y_0, x_1, y_1, \ldots, x_n, y_n)$ of the equal sized inner neighbors around the query block , i.e.,

$$ {}^iN = (\underline{x}_0, \underline{y}_0, \underline{x}_1, \underline{y}_1, \ldots \underline{x}_n, \underline{y}_n, \underline{x}_{n+1}, \underline{y}_{n+1}, \underline{x}_{n+2}, \underline{y}_{n+2}, \ldots \underline{x}_{n+k}, \underline{y}_{n+k}), \tag{23} $$

$$\underline{x}_j = x_j, \quad \underline{y}_j = y_j, \quad 0 \le j \le n, \tag{38}$$

$$\underline{x}_{n+h} = comp(x_n), \quad \underline{y}_{n+h} = comp(y_n), \quad h \ge 1. \tag{39}$$

For example, block 120 in Figure 6-(a) has an smaller sized inner neighbor 1230 in the diagonal direction which is derived from the equal sized inner neighbor 123 in the diagonal direction (in Figure 5-(a)) by using equations (38) and (39).

For the smaller sized outer neighbors at the $(n+k)$th level in the diagonal direction, it can be classified into three cases.

1. x-outer neighbor's (along x-axis) smaller sized y-inner neighbor which is along y-axis, e.g., block 0331 in Figure 6-(a).

2. y-outer neighbor's (along y-axis) smaller sized x-inner neighbor which is along x-axis, e.g., block 1032 in Figure 6-(a).

3. x-outer neighbor's (along x-axis) smaller sized y-outer neighbor which is along y-axis or y-outer neighbor's (along y-axis) x-outer smaller sized neighbor which is along x-axis, e.g., block 0133 in Figure 6-(a).

First, let's consider the case of x-outer neighbor's (along x-axis) smaller sized y-inner neighbor which is along y-axis. Its binary locational code $(\{\underline{x}_i, \underline{y}_i\},\ 0 \le i \le n+k)$ is derived from the code $(x_0, y_0, x_1, y_1, \ldots, x_n, y_n)$ of the equal sized x-outer neighbor around the query block where

$$\underline{x}_j = x_j, \quad \underline{y}_j = y_j, \quad 0 \le j \le n-1, \tag{40}$$

$$\underline{x}_n = x_n, \quad \underline{y}_n = comp(y_n), \quad \underline{x}_{n+h} = x_n, \quad \underline{y}_{n+h} = y_n, \quad h \ge 1. \tag{41}$$

Second, let's consider the case of y-outer neighbor's (along y-axis) smaller sized x-inner neighbor which is along x-axis. Its binary locational code $(\{\underline{x}_i, \underline{y}_i\},\ 0 \le i \le n+k)$ is derived from the code $(x_0, y_0, x_1, y_1, \ldots, x_n, y_n)$ of the equal sized y-outer neighbor around the query block where

$$\underline{x}_j = x_j, \quad \underline{y}_j = y_j, \quad 0 \le j \le n, \tag{42}$$

$$\underline{x}_n = comp(x_n), \quad \underline{y}_n = y_n, \quad \underline{x}_{n+h} = x_n, \quad \underline{y}_{n+h} = y_n, \quad h \geq 1. \tag{43}$$

Third, let's consider the case of x-outer neighbor's (along x-axis) smaller sized y-outer neighbor which is along y-axis , or y-outer neighbor's (along y-axis) smaller sized x-outer neighbor which is along x-axis. Its binary locational code ($\{\underline{x}_i, \underline{y}_i\}$, $0 \leq i \leq n+k$) is derived from the code of the equal sized x-outer neighbor's (along x-axis) y-outer neighbor which is along y-axis. Its code also can be derived from the code of the equal sized y-outer neighbor's (along y-axis) x-outer neighbor which is along x-axis. Assume the code of equal sized outer neighbor is $(x_0, y_0, x_1, y_1, \ldots, x_n, y_n)$, the equations between these code are shown as follows.

$$\underline{x}_j = x_j, \quad \underline{y}_j = y_j, \quad 0 \leq j \leq n, \tag{44}$$

$$\underline{x}_{n+h} = x_n, \quad \underline{y}_{n+h} = y_n, \quad h \geq 1. \tag{45}$$

For example, block 120 in Figure 6-(a) has three smaller sized outer neighbors, including block 0331 which is derived from the equal sized outer neighbor 031 by using equations (40) and (41), block 1032 which is derived from the equal sized outer neighbor 102 by using equations (42) and (43), and block 0133 which is derived from the equal sized outer neighbor 013 by using equations (44) and (45).

Next, we illustrate how to use these equations (7) – (45) for different sized neighbors finding of blo ck 022, block 323, block 222 shown in Figure 5-(b), (c), (d), and Figure 6-(b), (c), (d), which are the degenerated cases since some neighbors cannot be found.

- **Example 1.**For neighbor finding of block 022 in Figure 5-(b) and Figure 6-(b), its binary form of the locational code is: $(\underline{x_0, y_0}, \underline{x_1, y_1}, \underline{x_2, y_2}) = (\underline{0, 0}, \underline{0, 1}, \underline{0, 1})$.

  1. *Equal sized inner neighbors in $D = [1, 0]$ (along x-axis) and in $D = [0, 1]$ (along y-axis):*
     Based on Voros's strategy [13], we can obtain two inner neighbors: $(\underline{0, 0}, \underline{0, 1}, \underline{1^*, 1}) =$ (023) in $D = [1, 0]$ and $(\underline{0, 0}, \underline{0, 1}, \underline{0, 0^*}) =$ (020) in $D = [0, 1]$.
  2. *Equal sized outer neighbors in $D = [1, 0]$ and in $D = [0, 1]$:*
     Based on Voros's strategy [13], there exists no outer neighbor in $D = [1, 0]$ and exists one outer neighbor $(\underline{0, 1^*}, \underline{0, 0^*}, \underline{0, 0^*}) =$ (200) in $D = [0, 1]$.

3. *Equal sized inner neighbors in the diagonal direction:*

   According to equations (7), (8), (27), and (28), we can obtain the inner neighbor $(\underline{0,0},\underline{0,1},\underline{1^*,0^*}) = (021)$ in the diagonal direction.

   The inner neighbors of block 022 is summarized as follows

direction	binary form	quaternary digits
$D = [1,0]$	$\underline{0,0},\underline{0,1},\underline{1^*,1}$	023
$D = [0,1]$	$\underline{0,0},\underline{0,1},\underline{0,0^*}$	020
Diagonal	$\underline{0,0},\underline{0,1},\underline{1^*,0^*}$	021

4. *Equal sized outer neighbors in the diagonal direction:*

   According to equation (11), the first outer neighbor does not exist, because k $(= 3)$ is greater than n $(= 2)$. Following equations (15), (16), (17), (31), and (32), the second outer neighbor $(\underline{0,1^*},\underline{0,0^*},\underline{1^*,0^*}) = (201)$ is obtained. But the third outer neighbor does not exist, because h $(= 3)$ is greater than n $(= 2)$ according to equation (33).

   The outer neighbors of block 022 is summarized as follows

direction	binary form	quaternary digits
$D = [1,0]$	-	-
$D = [0,1]$	$\underline{0,1^*},\underline{0,0^*},\underline{0,0^*}$	200
Diagonal	$\underline{0,1^*},\underline{0,0^*},\underline{1^*,0^*}$	201

   There are five equal sized neighbors around block 022 shown in the Figure 5-(b).

5. *Larger sized outer neighbors:*

   Based on Voros's strategy [13], there exists larger sized outer neighbors $(\underline{2}\ \underline{0})$ and $(\underline{2})$ which are derived from the equal sized outer neighbor $(\underline{2}\ \underline{0}\ \underline{0})$.

6. *Smaller sized neighbors:*

   Based on Voros's strategy [13], the smaller sized neighbors are shown as follows.

direction	smaller sized neighbor	the corresponding equal sized neighbor
$D = [1,0]$	0230 and 0232	023
$D = [0,1]$	0202 and 0203	022
	2000 and 2001	200

From equations (38)–(45), the smaller sized neighbors in the diagonal direction are shown as follows.

direction	smaller sized neighbor	the corresponding equal sized neighbor
Diagonal	0212	021
	2010	200

- **Example 2.** For neighbor finding of block 323 in Figure 5-(c) and Figure 6-(c), its binary form of the locational code is $(\underline{x_0, y_0}, \underline{x_1, y_1}, \underline{x_2, y_2}) = (\underline{1, 1}, \underline{0, 1}, \underline{1, 1})$.

  1. *Equal sized inner neighbors in $D = [1, 0]$ and in $D = [0, 1]$:*

     Based on Voros's strategy [13], we can obtain two inner neighbors: $(\underline{1, 1}, \underline{0, 1}, \underline{0^*, 1}) =$ (322) in $D = [1, 0]$ and $(\underline{1, 1}, \underline{0, 1}, \underline{1, 0^*}) =$ (321) in $D = [0, 1]$.

  2. *Equal sized outer neighbors in $D = [1, 0]$ and in $D = [0, 1]$:*

     Based on Voros's strategy [13], there exists no outer neighbor in $D = [0, 1]$ and exists one outer neighbor $(\underline{1, 1}, \underline{1^*, 1}, \underline{0^*, 1}) =$ (332) in $D = [1, 0]$.

  3. *Equal sized inner neighbors in the diagonal direction:*

     According to equations (7), (8), (27), and (28), we can obtain the inner neighbor $(\underline{1, 1}, \underline{0, 1}, \underline{0^*, 0^*}) =$ (320) in the diagonal direction.

     The inner neighbors of block 323 is summarized as follows

direction	binary form	quaternary digits
$D = [1, 0]$	$\underline{1, 1}, \underline{0, 1}, \underline{0^*, 1}$	322
$D = [0, 1]$	$\underline{1, 1}, \underline{0, 1}, \underline{1, 0^*}$	321
Diagonal	$\underline{1, 1}, \underline{0, 1}, \underline{0^*, 0^*}$	320

  4. *Equal sized outer neighbors in the diagonal direction:*

     According to equations (11), (12), (13), (29), and (30), the first outer neighbor $(\underline{1, 1}, \underline{1^*, 1}, \underline{0^*, 0^*}) =$ (330) is obtained. According to equation (15), the second outer neighbor does not exist, because k $(= 3)$ is greater than n $(= 2)$. But the third outer neighbor does not exist, because k $(= 3)$ is greater than n $(= 2)$ according to equation (33).

     The outer neighbors of block 323 is summarized as follows

direction	binary form	quaternary digits
$D = [1,0]$	$\underline{1,1},\underline{1^*,1},\underline{0^*,1}$	332
$D = [0,1]$	-	-
Diagonal	$\underline{1,1},\underline{1^*,1},\underline{0^*,0^*}$	330

There are five equal sized neighbors around block 323 shown in the Figure 5-(c).

5. *Larger sized outer neighbors:*

   Based on Voros's strategy [13], there exists larger sized outer neighbors ($\underline{3}$ $\underline{3}$) which is derived from the equal sized outer neighbor ($\underline{3}$ $\underline{3}$ $\underline{2}$).

6. *Smaller sized neighbors:*

   Based on Voros's strategy [13], the smaller sized neighbors are shown as follows.

direction	smaller sized neighbor	the corresponding equal sized neighbor
$D = [1,0]$	3221 and 3223	322
	3320 and 3322	332
$D = [0,1]$	3212 and 3213	321

   From equations (38)–(45), the smaller sized neighbors in the diagonal direction are shown as follows.

direction	smaller sized neighbor	the corresponding equal sized neighbor
Diagonal	3302	332
	3203	320

- **Example 3.** For neighbor finding of block (222) in Figure 5-(d) and Figure 6-(d), its binary form of the locational code is $(\underline{x_0, y_0}, \underline{x_1, y_1}, \underline{x_2, y_2}) = (\underline{0,1}, \underline{0,1}, \underline{0,1})$.

1. *Equal sized inner neighbors in $D = [1,0]$ and in $D = [0,1]$:*

   Based on Voros's strategy [13], we can obtain two inner neighbors: $(\underline{0,1},\underline{0,1},\underline{1^*,1}) =$ (223) in $D = [1,0]$ and $(\underline{0,1},\underline{0,1},\underline{0,0^*}) =$ (220) in $D = [0,1]$.

2. *Equal sized outer neighbors in $D = [1,0]$ and in $D = [0,1]$:*

   Based on Voros's strategy [13], there exists no outer neighbors in $D = [0,1]$ and in $D = [1,0]$.

3. *Equal sized inner neighbors in the diagonal direction:*

   According to equations (7), (8), (27), and (28), we can obtain the inner neighbor $(\underline{0,1},\underline{0,1},\underline{1^*,0^*}) = (221)$ in the diagonal direction.

   The inner neighbors of block 222 is summarized as follows

direction	binary form	quaternary digits
$D = [1,0]$	$\underline{0,1},\underline{0,1},\underline{1^*,1}$	223
$D = [0,1]$	$\underline{0,1},\underline{0,1},\underline{0,0^*}$	220
Diagonal	$\underline{0,1},\underline{0,1},\underline{1^*,0^*}$	221

4. *Equal sized outer neighbors in the diagonal direction:*

   According to equation (11), the first outer neighbor does not exist, because k $(= 3)$ is greater than n $(= 2)$. According to equation (15), the second outer neighbor does not exist, because k $(= 3)$ is greater than n $(= 2)$. But the third outer neighbor does not exist either, because h $(= 3)$ and k $(= 3)$ are both greater than n $(= 2)$ according to equation (33). So, there are three equal sized neighbors, only the inner neighbors, around block 222 shown in Figure 5-(d).

5. *Larger sized outer neighbors:*

   Based on Voros's strategy [13], there exists no larger sized outer neighbor.

6. *Smaller sized neighbors:*

   Based on Voros's strategy [13], the smaller sized neighbors are shown as follows.

direction	smaller sized neighbor	the corresponding equal sized neighbor
$D = [1,0]$	2230 and 2232	223
$D = [0,1]$	2202 and 2203	220

   From equations (38)–(45), the smaller sized neighbors in the diagonal direction are shown as follows.

direction	smaller sized neighbor	the corresponding equal sized neighbor
Diagonal	2212	221

5 Conclusion

The strategy of neighbor finding has been proposed by Jozef Voros [13] to obtain the inner and outer neighbors in the quadtrees. However, missing cases would happen when the nearest neighbors locate in the diagonal direction and they can not be found by Voros's strategy. In this paper, we have shown such a case and have provided an answer to solve the missing case.

References

[1] Alberto Belussi, Elisa Bertino, and Barbara Catania, "Using Spatial Data Access Structures for Filtering Nearest Neighbor Queries," *Data and Knowledge Engineering*, Vol. 40, No. 1, pp. 1–31, 2002.

[2] Volker Gaede and Oliver Gunther, "Multidimensional Access Methods," *ACM Computing Surveys,* Vol. 30. No. 2, pp. 123-169, 1998.

[3] Irene Gargantini, "An Effective Way to Represent Quadtrees," *Comm. of ACM*, Vol. 12, No. 25, pp. 905–910, Dec. 1982.

[4] Ralf Hartmut Guting, "An Introduction to Spatial Database Systems," *Specail Issue on Spatial Database Systems of VLDB Journal*, Vol. 3, No. 4, pp. 1–32, Oct. 1994.

[5] Raghu Ramakrishnan and Johannes Gehrke, "Database Management Systems," *McGraw-Hill*, Aug. 1999.

[6] Hanan Samet, "Neighbor Finding Techniques for Images Represented by Quadtrees," *Comput. Graphics Image Process*, Vol. 18, No. 1, pp. 37–57, Jan. 1982.

[7] Hanan Samet, "Neighbor Finding Techniques for Images Represented by Octrees," *Comput. Vision Graphics Image Process*, Vol. 46, No. 1, pp. 367–386, Jan. 1989.

[8] Hanan Samet, "The Design and Analysis of Spatial Data Structures," *Addison-Wesley, Reading, MA*, 1990.

[9] Hanan Samet, "Applications of Spatial Data Structures: Computer Graphics, Image Processing, and Gis," *Addison-Wesley, Reading, MA*, 1990.

[10] Hanan Samet, "Spatial Data Structure," *Modern Database Systems: The Object Model, Interoperability, and Beyond,* W. Kim, ed., Addison Wesley/ACM Press, Reading, MA, pp. 361-385, 1994.

[11] Gunther Schback, "Finding Neighbors of Equal Size in Linear Quadtrees and Octrees in Constant Time," *CVGIP: Image Understanding*, Vol. 35, No. 3, pp. 221–230, 1992.

[12] Shashi Shekhar, Sanjay Chawla, Siva Ravada, Andrew Fetterer, Xuan Liu, and Chang-Tien Lu,"Spatial Databases — Accomplishments and Research Needs," *IEEE Trans. on Knowledge And Data Engineering*, Vol. 11, No. 1, pp. 45–55, Jan. 1999.

[13] Jozef Voros, "A Strategy for Repetitive Neighbor Finding in Images Represented by Quadtrees," *Pattern Recognition Letters*, Vol. 18, No. 10, pp. 955–962, Oct. 1997.

[14] Yuh-Horng Yang, Kuo-Liang Chung, and Yao-Hong Tsai, "A Compact Improved Quadtree Representation with Image Manipulations," *Image and Vision Comput.*, Vol. 18, No. 3, pp. 223–231, 1999.